# Podcast Episode

Public Sector
Embracing Digital Transformation

# Confidential Computing in the DevSecOps

Darren W Pulsipher & Ofir Azoulay-Rozanes – May 19, 2022

## Recent cyberattacks have focused on the build process vulnerabilities. Find out how to secure the DevOps pipeline to protect from these attacks with Confidential Computing.

In this episode, Darren Pulsipher, Chief Solutions Architect, Intel, and Ofir Azoulay-Rozanes, Director of Product Management, Anjuna, discuss Anjuna's solutions for confidential computing in the DevOps lifecycle.

**Video**: Youtube Channel

**Podcast**: SoundCloud

**Keywords:**
EDT87,EmbracingDigital,ConfidentialComputing,IntelPublicSector, DevSecOps

Anjuna's software enables applications to run with Intel's SGX protection and solves the problem of protecting data in use. Anjuna's mission is to make secure enclaves as simple as possible. With Anjuna's software, there is no need to change anything in the application; take it, run it in an enclave, and the SGX technology will work out of the box. The software works with any app, in any cloud, at any scale.

The global software supply chain is under attack. SolarWinds, notably, was an attack on DevOps, and although there have been ideas on how to solve the problem, it hasn't been locked down. Anjuna technology can be an easy solution. There is no need to re-architect your software or change methodologies. You run them in secure enclaves.

When trying out Anjuna's software, Darren created a stack using Intel SGX on the bottom, Red Hat OpenShift, Anjuna for the confidential computing part, and HashiCorp's Vault to store a secure ledger. He was shocked at how fast the solution was up and running in less than a week.

Darren calls this process the hardened DevSecOps pipeline, although it's many moving parts. Ofir agrees with this terminology, as this process is a new DevOps hardened with the SGX hardware technology with Anjuna's software.

Confidential computing, or secure enclave, solves the problem of protecting data. When you store data in persistent storage, the solution for data at rest is already there. There is also a solution for data in transit with TLS. Securing data in use has not been solved because when data is in use, the application needs to access it from memory in the clear. It can't be both encrypted and in use at the same time. This has been an endless loop of a problem. If a bad player has access to a machine where the application is running, a hack is as simple as coming through the device, identifying the process, and creating a memory dump. They will get all the secrets and confidential data on file, and it's not encrypted. This would also include the keys to encryption for data at rest and in transport because the software needs to use them to encrypt. The lousy player will have the keys to the kingdom.

The problem is resolved if you run the different applications in secure enclaves. Even if someone got access to the machine, they would not have access to the memory of each application. This doesn't mean you don't have to resolve vulnerabilities, but you are much less stressed to fix them as soon as possible. Even if there are kernel vulnerabilities, when something runs in a secure enclave, the kernel cannot access its memory.

Anjuna software runs on other hardware-based technologies as well as SGX. Unlike running encryption in software, where the performance hit would be high, Anjuna can fine-tune the configuration to run your application with a negligible performance hit: less than five percent.

So you may not want to put everything in a secure enclave just yet, but it is the future for security.

One of the uses for a secure enclave is to store data that pans different steps in the DevOps pipeline in a secure ledger. The ledger has everything that went into the build, security keys, and hash values used for verification. These verification hash must remain unchanged through the cycle so no one can inject code, libraries, or binaries into the package you deliver. Everything should run in a container in the modern world.

Another candidate for protection is a signing key. Without secure enclaves, once you have a binary ready, you need to take it to another machine in a dark room that no one has access to. But three people with three different keys sign it there. Secure enclaves enable access to that signing key in your familiar environment, but only the enclave will access it. It will be based on the complex identity of the software running inside the SGX enclave, which is implemented via the attestation quote. In other words, you can attest enclave to enclave. You can also attest to things that run outside of enclaves. It gives you the ability to trust software that runs somewhere else.

The compilation of binaries is another use. One of the big problems in the Department of Defense, for example, is that they want to be guaranteed that everything that went into the build can be traced back to the developer who wrote it. Especially in embedded systems where software controls multimillion-dollar machines that can kill people or save people's lives. There must be full traceability to help ensure accountability and secure development has been performed.

In addition to memory dump attacks, another attack problem that Anjuna solves is making sure that in cases where you need to go to the kernel, it will protect whatever needs to be covered in that interaction between the enclave and the outside world. It also can protect against accessing code and make secrets only available to the enclave. In addition, if someone gets into a machine, they won't be able to find a TLS certificate in the clear or the key that's used to encrypt it.

Every cloud service provider offers secure enclaves, and Anjuna supports them all. They also support on-prem technologies. On top of the primary offering, Anjuna can also enable the ability to encrypt your data at rest and in transit without changing your software, even in legacy applications or new applications that don't support encrypting every data file.

For more information, visit anjuna.io, or check out a white paper authored by Darren and Ofir at embracingdigital.com.

.